

Uvod v Mathematico

Nekaj starega, nekaj novega

Osnovna uporaba

Vgrajene funkcije

Simbolicno racunanje

Seznamni

Stevila

Nizi

Spremenljivke in funkcije

Prepisovalna pravila

Grafika

Interaktivnost

If, for, while

Slovarji

```
In[1]:= mojaShramba = <| "jabolko" -> 6, "banana" -> 5 |>
Out[1]= <| jabolko -> 6, banana -> 5 |>
```

```
In[2]:= mojaShramba["jabolko"]
```

```
Out[2]=
```

```
In[1]:= mojaShramba["kivi"]
Out[1]= Missing[KeyAbsent, kivi]

In[2]:= mojaShramba["jabolko"] = 7
mojaShramba["kivi"] = 1

Out[2]= 7

Out[3]= 1

In[4]:= mojaShramba
Out[4]= <| jabolko → 7, banana → 5, kivi → 1 |>

In[5]:= TemplateApply["V shrambi imamo `jabolko` jabolk in `banana` banan.", mojaShramba]
Out[5]= V shrambi imamo 7 jabolk in 5 banan.
```

Vnos z naravnim jezikom

Nacini vnosa

V novejsih razlicicah Mathematica omogoca, da poiscemo informacije o dolocenem objektu ali celo izvedemo (krajsi) izracun kar z naravnim jezikom. Vnos zacnemo z =.

```
In[1]:= Slovenia COUNTRY
Out[1]= Slovenia

In[2]:= Triglav MOUNTAIN [ image ] ✓
Out[2]=
```



Ob vnosu == dobimo porocilo iz WolframAlpha.

```
In[3]:= Triglav
```

In[1]:=

 temperature in Ljubljana in last 10 days
In[2]:=  compute integral of x^2 from -3 to 10

Out[2]=

$$\frac{1027}{3}$$

Objekte, ki jih dobimo z vnosom z naravnim jezikom, lahko shranimo v spremenljivke in z njimi racunamo. Mathematica ima podatke o drzavah, mestih, kemikalijah, zivalskih in rastlinskih vrstah, filmih, ljudeh, letaliscih ... Objekt dobimo z **CTRL + =**.

In[3]:= **Ljubljana** =  Ljubljana CITY

Out[3]=

 LjubljanaIn[4]:=  Ljubljana CITY ["Properties"]

Out[4]=

$$\left\{ \begin{array}{l} \text{active home listings}, \text{administrative region}, \\ \text{number of aggravated assaults}, \text{rate of aggravated assault}, \\ \text{aggregate home value}, \text{aggregate home value, householder 15 to 24 years}, \\ \text{aggregate home value, householder 25 to 34 years}, \\ \text{aggregate home value, householder 35 to 64 years}, \\ \text{aggregate home value, householder 65 years and over}, \text{aggregate household income}, \\ \text{airport codes}, \text{fuel spent in delays}, \text{total fuel spent in delays}, \text{average annual delay}, \\ \text{total annual delay}, \text{area}, \text{area code}, \text{arterial street traffic}, \text{arterial street length}, \\ \text{average home listing price}, \text{average public transit trip distance}, \text{number of burglaries}, \\ \text{rate of burglary}, \text{city sales tax}, \text{coordinates}, \text{country}, \text{county}, \text{county sales tax}, \\ \text{total rate of crime}, \text{total number of crimes}, \text{average daily traffic delay}, \\ \text{elevation}, \text{entity classes}, \text{number of rapes}, \text{rate of rape}, \text{freeway traffic}, \\ \text{freeway length}, \text{Gini index}, \text{has polygon?}, \text{FHFA home price index}, \\ \text{FHFA home price index annual average}, \text{housing affordability index}, \\ \text{home listings with increased price}, \text{home listings with reduced price}, \text{households}, \\ \text{number of larcenies}, \text{rate of larceny}, \text{latitude}, \text{longitude}, \text{total magnetic field strength}, \\ \text{median age}, \text{median home listing time on market}, \text{median home listing price}, \\ \text{median home listing price per square foot}, \text{median home sale price}, \text{median home size}, \\ \text{median home value}, \text{median household income}, \text{number of motor vehicle thefts} \end{array} \right.$$

rate of motor vehicle thefts , incidents of murder and nonnegligent manslaughter ,
 rate of murder and nonnegligent manslaughter , name , new home listings ,
 next daylight saving shift , nicknames , number of owner-occupied housing units ,
 average daily peak period travelers , peak period travelers per capita ,
 average daily peak period vehicles , pending home listings , notable people born in city ,
 notable people who died in city , per capita income , polygon , city population ,
 population by educational attainment , population by migration in previous 12 months ,
 population by language spoken at home , population by marital status ,
 population by poverty status , population by school enrollment , population density , position ,
 last daylight saving shift , total rate of property crime , total number of property crimes ,
 total public transit use , unlinked public transit trips , 4 bedroom apartment fair market rent ,
 1 bedroom apartment fair market rent , 3 bedroom apartment fair market rent ,
 2 bedroom apartment fair market rent , studio apartment fair market rent ,
 number of robberies , rate of robbery , daily rush hour length , state sales tax ,
 time zone , total daily traffic delay , total sales tax rate , average peak travel time ,
 unemployment rate , unweighted sample housing units , unweighted sample population ,
 total rate of violent crime , total number of violent crimes , ZIP codes }
}

In[]:= CanonicalName[average home listing price]

Out[]=

AverageHomeListingPrice

In[]:= ljubljana["Area"]

Out[]=

163.76 km²

Alternativno lahko objekt dobimo s funkcijo **EntityClass**.

In[]:= EntityClass["Country", "Europe"]

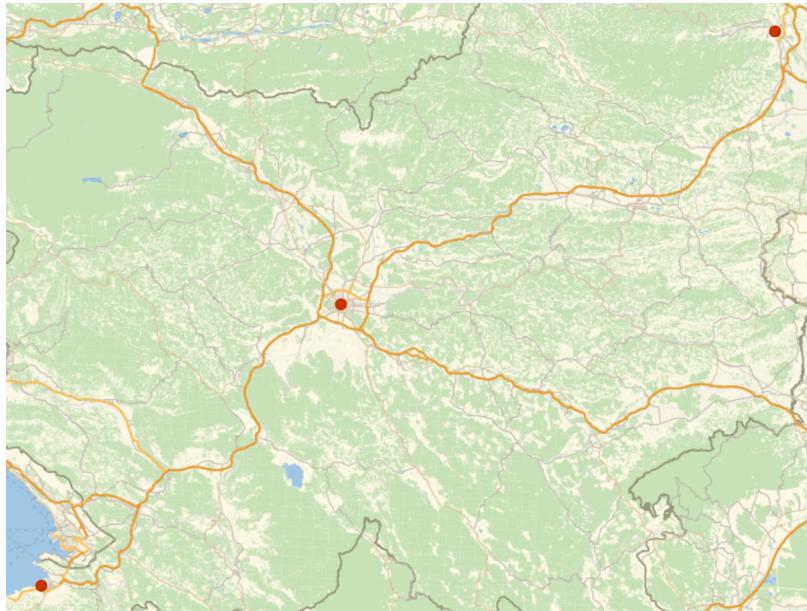
Out[]=

Europe

Geografsko risanje

```
In[1]:= GeoListPlot[{Ljubljana, Maribor CITY ..., Koper CITY ...}]
```

Out[1]=



Seznam vseh mest v Sloveniji, o katerih ima Mathematica informacije:

```
In[2]:= EntityClass["City", {"Country" → "Slovenia"}] // EntityList
```

Out[2]=

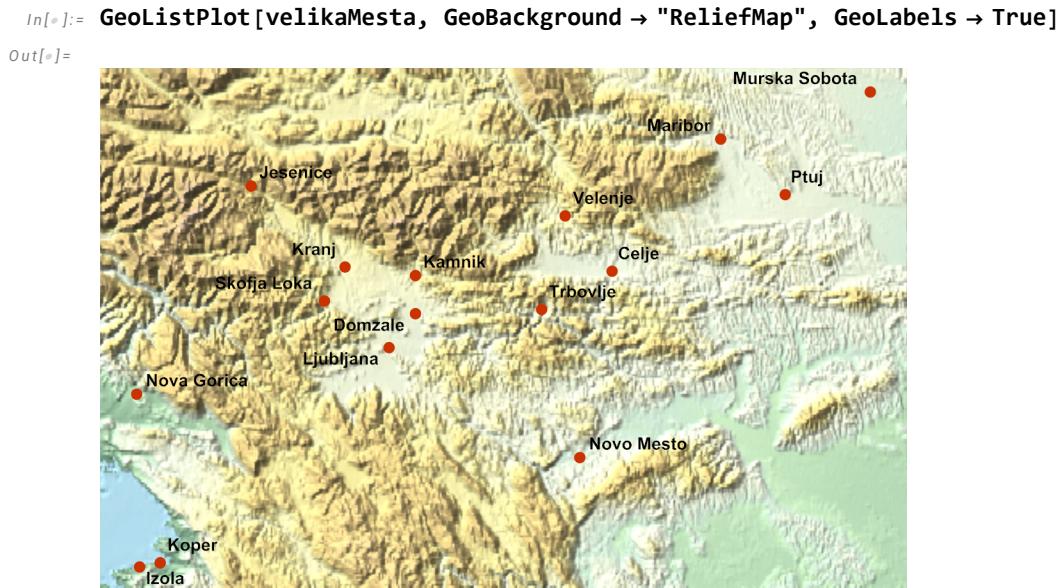
```
{Ljubljana, Maribor, Celje, Kranj, Velenje, Koper, Novo Mesto, Ptuj,
Trbovlje, Kamnik, Jesenice, Nova Gorica, Domzale, Skofja Loka,
Murska Sobota, Izola, Postojna, Kocevje, Logatec, Bled, Vrhnika,
Slovenj Gradec, Slovenska Bistrica, Krsko, Grosuplje, Ravne, Ajdovscina,
Brezice, Zagorje ob Savi, Litija, Menges, Idrija, Radovljica, Crnomelj,
Lucija, Hrastnik, Sezana, Rogaška Slatina, Medvode, Slovenske Konjice,
Zalec, Ilirska Bistrica, Sevnica, Sentjur, Ruse, Prevalje, Trzic, Piran,
Miklavz na Dravskem Polju, Sempeter, Tolmin, Ziri, Cerknica, Ribnica,
Trzin, Lasko, Ljutomer, Dravograd, Lendava, Solkan, Metlika, Ankaran,
Mezica, Trebnje, Bistrica ob Trzicu, Sencur, Vir, Gornja Radgona, Zelezniki,
Sostanj, Zrece, Portorož, Lesce, Radlje ob Dravi, Lenart, Limbuš,
Senovo, Muta, Vrtojba, Polzela, Crna na Koroskem, Brezovica pri Ljubljani,
Radece, Beltinci, Race, Koroska Bela, Jagodje, Radenci, Borovnica,
Spodnje Hoce, Ormoz, Pekre, Ig, Pivka, Straza, Lovrenc na Pohorju,}
```

Skofljica, Vojnik, Rakek, Mozirje, Sempeter, Slovenski Javornik, Store,
 Spodnja Idrija, Mislinja, Verd, Hrusica, Vnanje Gorice, Rence, Kromberk,
 Lavrica, Bohinjska Bistrica, Kisovec, Černelavci, Cerkno, Britof, Naklo,
 Bovec, Vuzenica, Smarje pri Jelsah, Kokrica, Odranci, Prebold, Radizel,
 Rogatec, Radomlje, Ivancna Gorica, Vipava, Bakovci, Zgornji Duplek, Rakičan,
 Kamnica, Mlaka pri Kranju, Turnisce, Mekinje, Dragomer, Notranje, Miren,
 Dol pri Hrastniku, Kranjska Gora, Šmartno pri Litiji, Mirna, Cerkle na Gorenjskem,
 Dekani, Razvanje, Dob, Log pri Brezovici, Sentilj, Smarje-Sap, Šmarca,
 Sentjernej, Bresternica, Sveti Anton, Oplotnica, Divaca, Kanal, Deskle,
 Preserje pri Radomljah, Bistrica ob Dravi, Selnica ob Dravi, Hotinja Vas, Pamece,
 Kidricevo, Smartno pri Slovenj Gradcu, Zgornje Bitnje, Kobarid, Mojstrana,
 Spodnje Škofije, Vodice, Topolšica, Crensovci, Rogoza, Prvačina, Horjul,
 Zgornje Pirniče, Ljubno ob Savinji, Gorenja Vas, Spodnji Duplek, Izlake,
 Poljocene, Gotovlje, Hrvatni, Sredisce ob Dravi, Zgornja Polskava, Pragersko,
 Bilje, Krog, Rožna Dolina, Trnovlje pri Celju, Šempas, Ravne, Hrastje,
 Zuzemberk, Selnica ob Muri, Lipovci, Golnik, Kotlje, Pobegi, Podgorje,
 Lokavec, Leskovac pri Krskem, Branik, Begunje na Gorenjskem, Spodnje Gorje,
 Gančani, Košaki, Zasip, Predoslje, Blejska Dobrava, Brestanica, Verzej,
 Bukovci, Nazarje, Podgorje, Prade, Dobrovnik, Ljubecna, Seča, Fram,
 Sveti Duh, Pareceg, Gornji Grad, Legen, Mirna Pec, Moravce, Zabukovica,
 Šentvid pri Stični, Dogoše, Dokležovje, Pristava, Petrovče, Dobrova,
 Velika Polana, Otiski Vrh, Bertoki, Dornava, Spodnja Besnica, Stojnici,
 Komenda, Lokovica, Kasaze, Vitanje, Petišovci, Škale, Preddvor,
 Kropa, Tolsti Vrh, Zadobrova, Stari trg pri Lozu, Rimske Toplice, Pobrežje,
 Stara Loka, Rovte, Visnja Gora, Zgornje Poljocene, Pesnica pri Mariboru,
 Spuhlja, Vuhred, Krčevina pri Vurbergu, Visoko, Melinci, Krize,
 Breg pri Polzeli, Spodnja Polskava, Skoke, Podvinci, Ižakovci, Loka pri Framu }

Seznam vseh velikih slovenskih mest:

```
In[1]:= velikaMesta = EntityClass["City",
  {"Country" → "Slovenia", "Population" → GreaterThan[10000]}] // EntityList
Out[1]= {Celje, Domžale, Izola, Jesenice, Kamnik, Koper,
Kranj, Ljubljana, Maribor, Murska Sobota, Nova Gorica,
Novo Mesto, Ptuj, Skofja Loka, Trbovlje, Velenje}
```

Velika mesta narisemo in poiscemo najkrajsi obhod po vseh:



```
In[3]:= vrstniRed = Last[FindShortestTour[GeoPosition[velikaMesta]]]
Out[3]= {1, 15, 12, 6, 3, 11, 4, 7, 14, 8, 2, 5, 16, 9, 10, 13, 1}
```

```
In[6]:= GeoListPlot[velikaMesta[[vrstniRed]], Joined -> True,
GeoLabels -> True, GeoRange -> Slovenia COUNTRY ]
```

Out[6]=



Temperatura

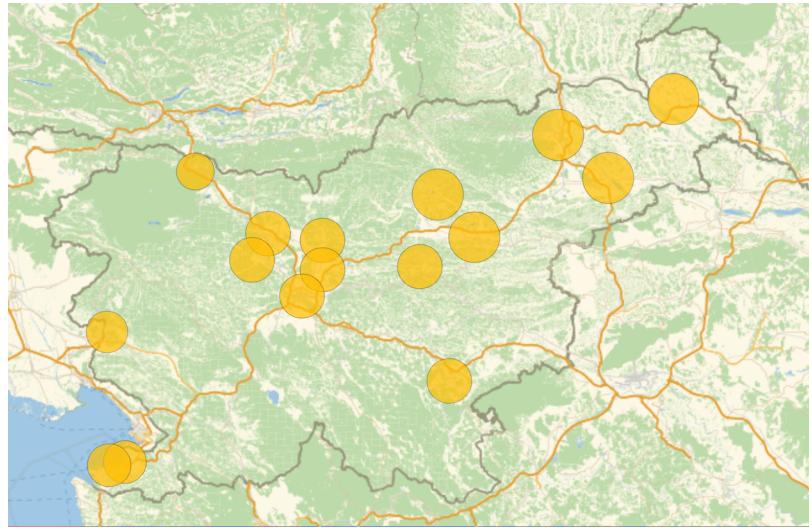
```
In[7]:= podatkiOTemperaturi = Table[m -> AirTemperatureData[m], {m, velikaMesta}]
```

Out[7]=

```
{Celje -> 17. °C, Domzale -> 13. °C, Izola -> 12. °C,
Jesenice -> 9. °C, Kamnik -> 13. °C, Koper -> 12. °C, Kranj -> 13. °C,
Ljubljana -> 13. °C, Maribor -> 17. °C, Murska Sobota -> 17. °C,
Nova Gorica -> 11. °C, Novo Mesto -> 13. °C, Ptuj -> 17. °C,
Skofja Loka -> 13. °C, Trbovlje -> 13. °C, Velenje -> 17. °C}
```

In[*#*]:= `GeoBubbleChart[podatkiOTemperaturi, GeoRange → Slovenia COUNTRY]`

Out[*#*]=



In[*#*]:= `AirTemperatureData[Ljubljana CITY, DateObject[{2020, 1}, "Month"], Min]`

Out[*#*]=

-5.8 °C

In[*#*]:= `minimalneTemperature[mesto_] := Table[{DateObject[{2020, i}, "Month"], AirTemperatureData[mesto, DateObject[{2020, i}, "Month"], Min]}, {i, 1, 12}]`

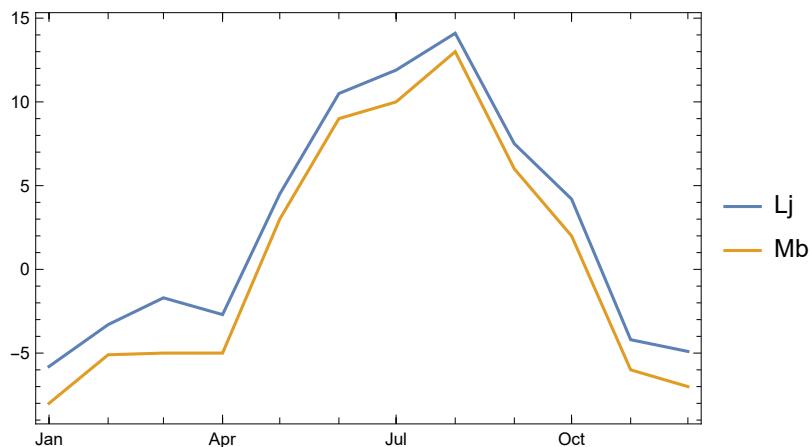
In[*#*]:= `minimalneTemperature[Ljubljana CITY ... ✓]`

Out[*#*]=

```
{ { Month: Jan 2020, -5.8 °C }, { Month: Feb 2020, -3.3 °C }, { Month: Mar 2020, -1.7 °C },
{ Month: Apr 2020, -2.7 °C }, { Month: May 2020, 4.5 °C }, { Month: Jun 2020, 10.5 °C },
{ Month: Jul 2020, 11.9 °C }, { Month: Aug 2020, 14.1 °C }, { Month: Sep 2020, 7.5 °C },
{ Month: Oct 2020, 4.2 °C }, { Month: Nov 2020, -4.2 °C }, { Month: Dec 2020, -4.9 °C } }
```

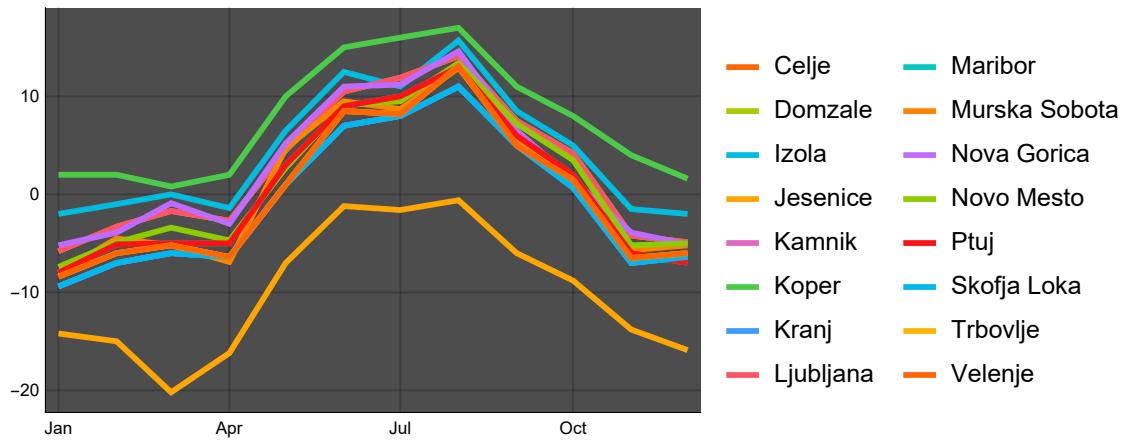
```
In[1]:= DateListPlot[{minimalneTemperature[Ljubljana CITY ...], minimalneTemperature[Maribor CITY ...]}, PlotLegends -> {"Lj", "Mb"}]
```

Out[1]=



```
In[2]:= DateListPlot[Table[minimalneTemperature[m], {m, velikaMesta}], PlotLegends -> Table[m["Name"], {m, velikaMesta}], PlotTheme -> "Marketing"]
```

Out[2]=



Indeks neenakosti

Indeks neenakosti med ljudmi merimo z Gini indeksom. Vecja vrednost pomeni vecjo neenakost.

```
In[3]:= Slovenia COUNTRY ["GiniIndex"]
```

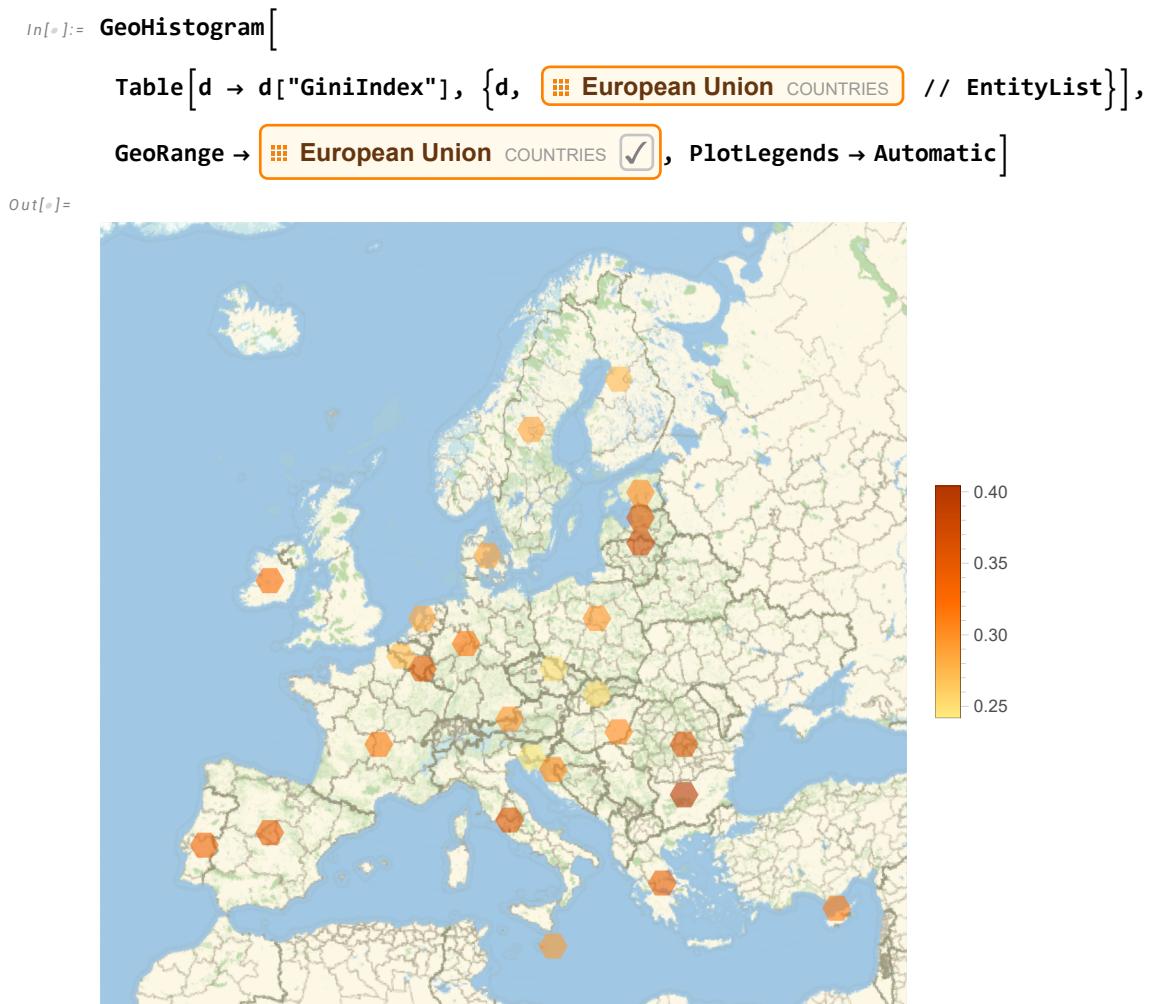
Out[3]=

0.244

```
In[4]:= #[ "GiniIndex"] & /@ (European Union COUNTRIES // EntityList)
```

Out[4]=

```
{0.297, 0.274, 0.404, 0.304, 0.314, 0.249, 0.287, 0.304,
0.274, 0.316, 0.319, 0.344, 0.306, 0.328, 0.359, 0.356, 0.373,
0.349, 0.292, 0.285, 0.297, 0.338, 0.360, 0.252, 0.242, 0.347, 0.288}
```



Obhod po vrocih spanskih mestih

Poiscimo najkrajsi obhod po vseh spanskih mestih z maksimalno temperaturo vec kot 40 stopinj avgusta lansko leto, vsako obiskano mesto naj ima vsaj 100,000 prebivalcev.

```
In[1]:= mesta = EntityClass["City", {"Country" → "Spain",
  "Population" → GreaterThan[⊖ 100 000 people ⋮ ✓]}] // EntityList
```

Out[1]=

```
{A Coruña, Albacete, Alcala de Henares, Alcobendas, Alcorcon, Algeciras,
Alicante, Almeria, Badajoz, Badalona, Barakaldo, Barcelona, Bilbao,
Burgos, Cadiz, Cartagena, Castello, Cordoba, Dos Hermanas, Elx,
Fuenlabrada, Getafe, Gijón, Granada, Huelva, Jaen, Jerez,
Las Palmas, Leganes, Leon, LHospitalet de Llobregat, Lleida, Logrono,
Madrid, Malaga, Marbella, Mataro, Mostoles, Murcia, Ourense,
Oviedo, Palma, Pamplona, Parla, Reus, Sabadell, Salamanca,
San Cristobal de la Laguna, San Sebastian, Santa Coloma de Gramenet,
Santa Cruz de Tenerife, Santander, Seville, Tarragona, Telde, Terrassa,
Torrejon de Ardoz, Valencia, Valladolid, Vigo, Vitoria, Zaragoza}
```

```
In[2]:= topLaMesta = Select[mesta,
```

```
AirTemperatureData[#, DateObject[{2020, 8}, "Month"], Max] > ⊖ 40 °C ⋮ ✓ &]
```

Out[2]=

```
{Cordoba, Dos Hermanas, Granada, Jaen,
Jerez, Malaga, Marbella, Murcia, Pamplona, Seville}
```

```
In[1]:= vrstniRed = Last[FindShortestTour[GeoPosition[toplaMesta]]];
GeoListPlot[toplaMesta[[vrstniRed]], Joined -> True,
GeoLabels -> True, GeoRange -> Spain COUNTRY ]
```

Out[1]=



Evropske drzave in prestolnice, ki so toplejse od Ljubljane
Temperamentne lastnosti mack

Manipuliranje in analiza slik

Osnovno manipuliranje slik

```
In[2]:= Directory[]
NotebookDirectory[]
```

Out[2]=

C:\Users\Tadej\Documents

Out[3]=

C:\Users\Tadej\OneDrive - Univerza v Ljubljani\Synced\Poucevanje\2023_pol_ROM\

```
In[]:= slika = Import[NotebookDirectory[] <> "background.png"]
```

```
Out[]=
```



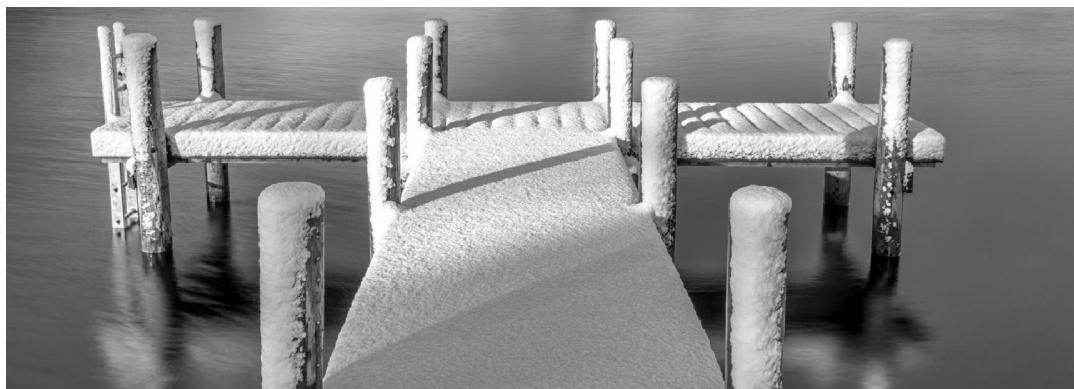
```
In[]:= DominantColors[slika]
```

```
Out[]=
```

```
{Blue, Cyan, DarkGray, Black, Brown, Yellow, Gold}
```

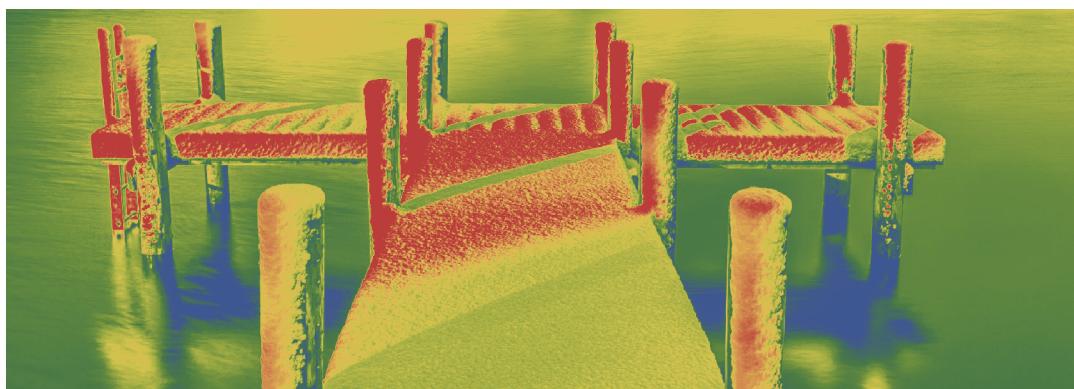
```
In[]:= ColorConvert[slika, "Grayscale"]
```

```
Out[]=
```



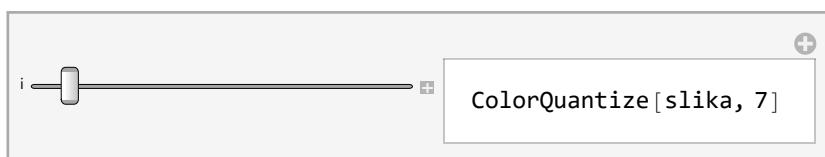
```
In[]:= Colorize[%]
```

```
Out[]=
```



```
In[]:= Manipulate[ColorQuantize[slika, i], {i, 1, 100, 1}]
```

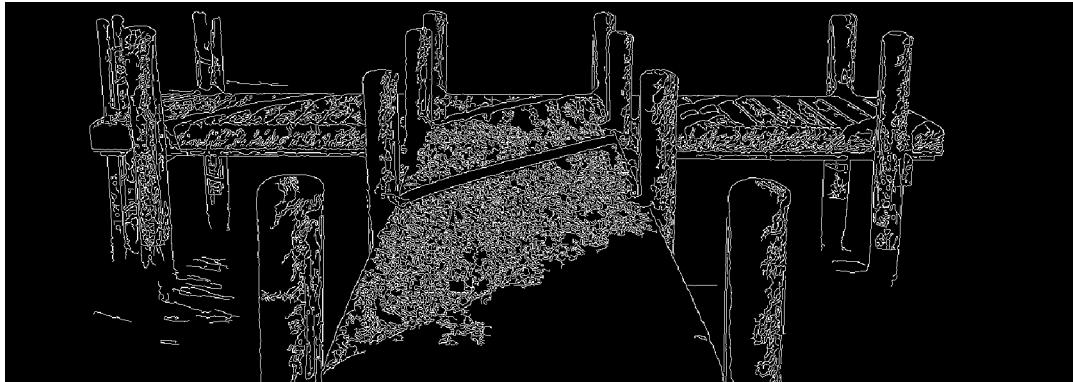
```
Out[]:=
```



ColorQuantize: Expecting an image or graphics instead of slika.

```
In[]:= EdgeDetect[slika]
```

```
Out[]:=
```



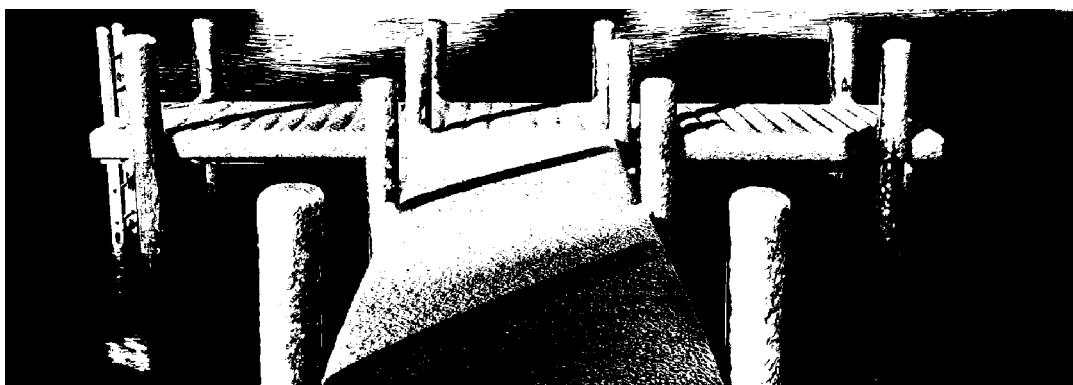
```
In[]:= ColorDetect[slika, Black]
```

```
Out[]:=
```



```
In[]:= Binarize[slika]
```

```
Out[]:=
```



```
In[1]:= Nearest[DominantColors[slika], Red, DistanceFunction → ColorDistance]
Out[1]= {■}
```

```
In[2]:= ColorReplace[slika, %[[1]] → Red]
Out[2]=
```



Prepoznavanje

```
In[1]:= s1 = Import[NotebookDirectory[] <> "dog1.jpg"];
s2 = Import[NotebookDirectory[] <> "dog2.jpg"];
s3 = Import[NotebookDirectory[] <> "dog3.jpg"];
s4 = Import[NotebookDirectory[] <> "human1.jpg"];
s5 = Import[NotebookDirectory[] <> "human2.jpg"];
s6 = Import[NotebookDirectory[] <> "human3.jpg"];
s7 = Import[NotebookDirectory[] <> "human4.jpg"];
s8 = Import[NotebookDirectory[] <> "who_am_I.jpg"];
s9 = Import[NotebookDirectory[] <> "who_am_I_2.jpg"];
s10 = Import[NotebookDirectory[] <> "who_am_I_3.jpg"];
s11 = Import[NotebookDirectory[] <> "who_am_I_4.jpg"];

In[2]:= trainingset = {s1 → "dog", s2 → "dog", s3 → "dog",
                     s4 → "human", s5 → "human", s6 → "human", s7 → "human"};
c = Classify[trainingset]
```

Out[2]=

ClassifierFunction[
 + Input type: Image
 Classes: dog, human]
 Data not in notebook. Store now

```
In[8]:= c[s8, "TopProbabilities"]
c[s9, "TopProbabilities"]
c[s10, "TopProbabilities"]
c[s11, "TopProbabilities"]

Out[8]= {dog → 1.}

Out[9]= {dog → 1.}

Out[10]= {dog → 1.}

Out[11]= {human → 1.}
```

Analiza slik

```
In[12]:= ImageContents[s1]
ImageContents[s10]

Out[12]=
```

Image	Concept	BoundingBox	Probability
	domestic dog	Rectangle[{347.14, 221.323}, {999.485, 907.752}]	0.848


```
Out[13]=
```

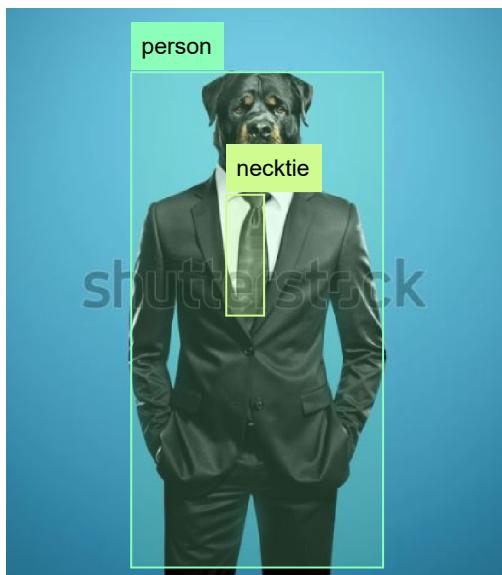
Image	Concept	BoundingBox	Probability
	person	Rectangle[{108.489, 7.5796}, {327.048, 437.421}]	0.89
	necktie	Rectangle[{190.952, 226.184}, {223.788, 331.067}]	0.74

```
In[14]:= skatle = ImageBoundingBoxes[s10]

Out[14]= {person → {Rectangle[{108.489, 7.5796}, {327.048, 437.421}]}, 
          necktie → {Rectangle[{190.952, 226.184}, {223.788, 331.067}]}}
```

```
In[]:= HighlightImage[s10, skatle]
```

```
Out[]=
```



```
In[]:= ImageContainsQ[s10, "person"]
```

```
Out[]=
```

```
True
```

```
In[]:= ImageContainsQ[s10, "necktie"]
```

```
ImageContainsQ[s10, "banana"]
```

```
Out[]=
```

```
True
```

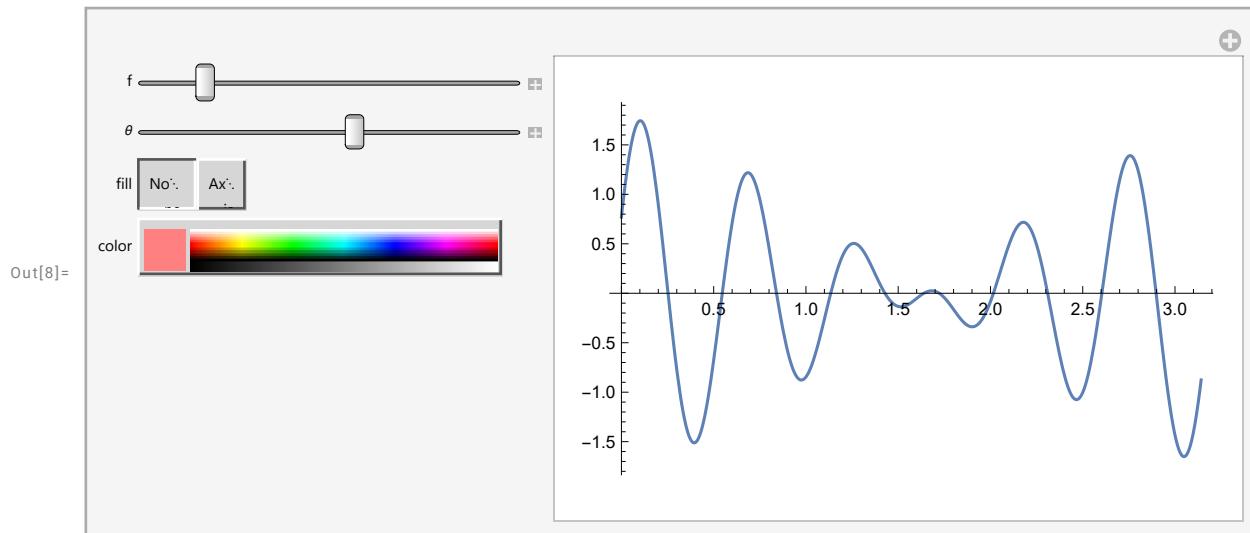
```
Out[]=
```

```
False
```

Oblak

V Mathematici lahko zelo hitro izdelamo preprosto aplikacijo/spletno stran z uporabo storitve *Wolfram Cloud*.

```
In[8]:= app = Manipulate[
 Plot[Sin[10 x] + Sin[f x + θ], {x, 0, π}, Filling → fill, FillingStyle → color],
 {f, 10, 20},
 {θ, 0, 4 π},
 {fill, {None, Axis}},
 {color, Pink}]
```



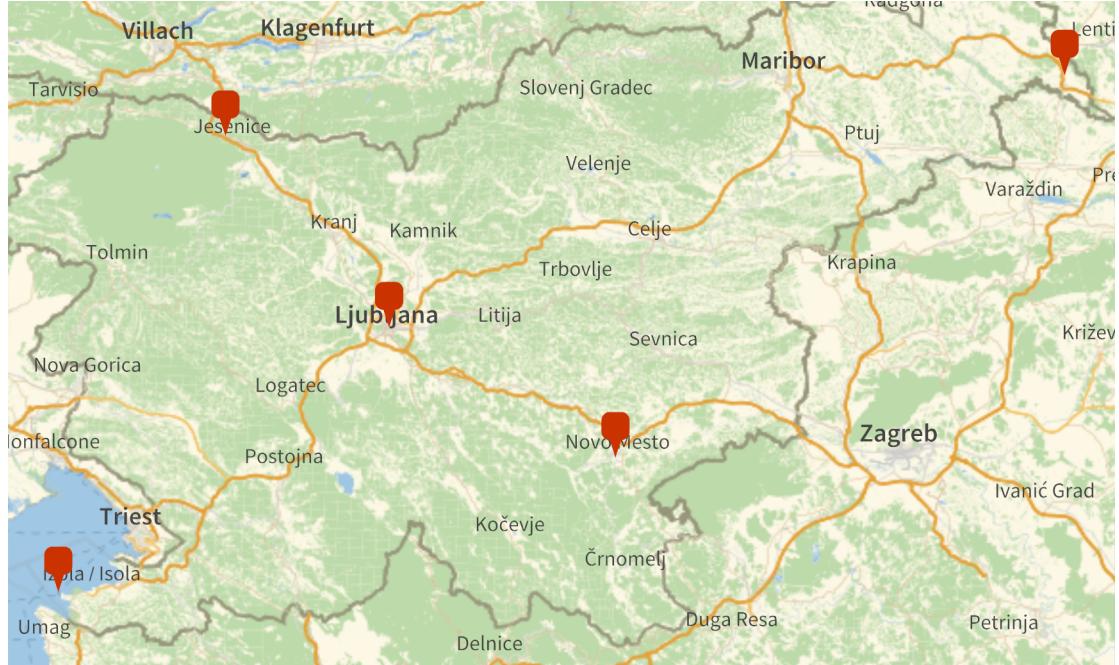
```
In[9]:= CloudDeploy[app, Permissions → "Public"]
Out[9]= CloudObject[https://www.wolframcloud.com/obj/6e794966-60c4-4d73-adf8-ef8aa4daaa1a]
```

še en primer....

```
In[10]:= kraji = {Ljubljana CITY ..., Jesenice CITY ..., 
  Piran CITY ..., Novo Mesto CITY ..., Lendava CITY ...};
```

GeoGraphics[GeoMarker /@ kraji, ImageSize → Large]

Out[11]=



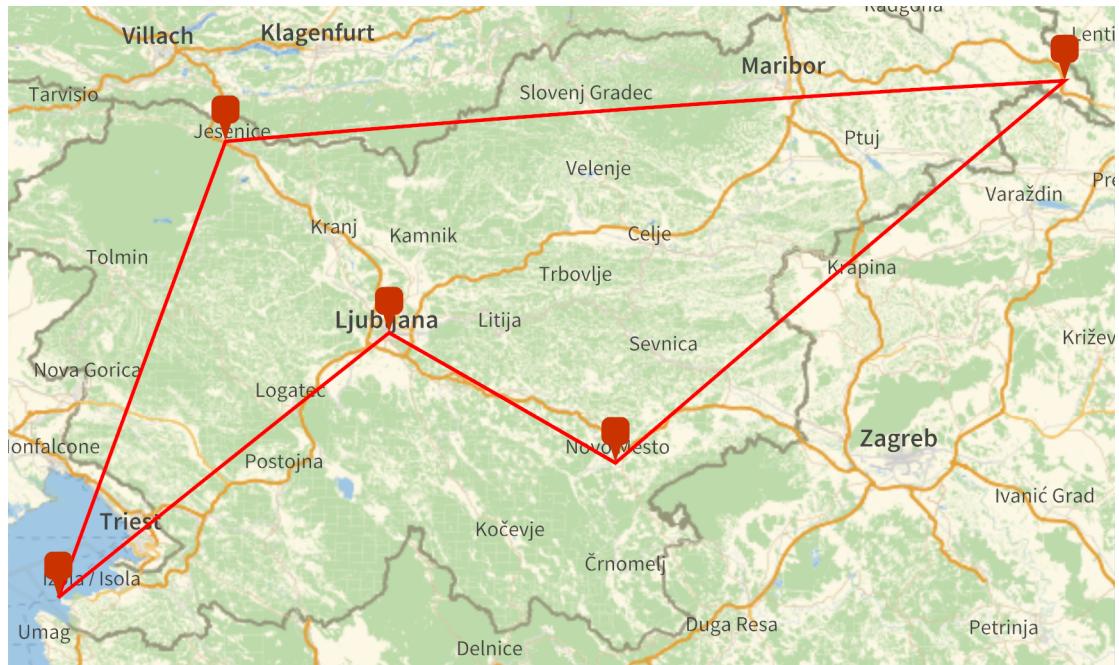
```
In[12]:= obhod = FindShortestTour[kraji]
```

Out[12]=

 $\{576.424 \text{ km}, \{1, 4, 5, 2, 3, 1\}\}$

```
In[13]:= GeoGraphics[
  {Thick, Red, GeoPath[kraji[[Last[obhod]]]], GeoMarker /@ kraji}, ImageSize → Large]
```

Out[13]=



```
In[1]:= Interpreter["Location"]["Jesenice"]
Out[1]= GeoPosition[{46.44, 14.04}]

In[13]:= NajkrajsiObhod[pozicije_] := Module[{obhod},
    obhod = FindShortestTour[pozicije];
    GeoGraphics[{Thick, Red,
        GeoPath[pozicije[[Last[obhod]]]], GeoMarker /@ pozicije}, ImageSize -> Large]
];
In[14]:= obrazec = FormPage["kraji" -> RepeatingElement["Location"],
    NajkrajsiObhod[#kraji] &,
    AppearanceRules -> <| "Title" -> "Najkrajsi obhod", "Description" ->
        "Vnesi imena krajev, ki jih zelis obiskati. Izracunal bom najkrajsi obhod.",
        "SubmitLabel" -> "Poisci obhod!" |>]
```

Out[14]=

Najkrajsi obhod

Vnesi imena krajev, ki jih zelis obiskati. Izracunal bom najkrajsi obhod.

kraji

-

+

Poisci obhod!

```
In[15]:= CloudDeploy[obrazec, Permissions -> "Public"]
Out[15]= CloudObject[https://www.wolframcloud.com/obj/466061bd-299e-4459-b9af-2365be388c1a]
```